REMARKS

Claims 1-15 are pending in the present continuation patent application. Applicant cancels claims 5-15. Applicant amends claim 1. Applicant adds claims 16-29. Consideration and examination of pending claims 1-4 and 16-29 is respectfully requested.

New Figures

Applicant encloses herein a set of formal drawings to replace the informal drawings filed with this continuation patent application. In the informal drawings, some of the figures were illustrated on a single drawing sheet (i.e., Figures 3, 4, 6, 7, 8, 9A, 10, and 12). To improve legibility, modifications were made to the informal drawings in preparing the formal drawings. Specifically, the following changes were made to the informal drawings in the preparation of the formal drawings:

(a) Figure 3 is split into two drawing sheets, Figures 3(1) and 3(2);

(b) Figure 4 is split into two drawing sheets, Figures 4(1) and 4(2);

(c) Figure 6 is split into two drawing sheets, Figures 6(1) and 6(2);

(d) Figure 7 is split into two drawing sheets, Figures 7(1) and 7(2);

(e) Figure 8 is split into two drawing sheets, Figures 8(1) and 8(2);

(f) Figure 9A is split into two drawing sheets, Figures 9A(1) and 9A(2);

(g) Figure 10 is split into two drawing sheets, Figures 10(1) and 10(2); and

(h) and Figure 12 is split into five drawing sheets, Figures 12(1), 12(2), 12(3), 12(4), and 12(5).

Accordingly, Applicants wish to amend the specification to change all "Figure 3" references to "Figures 3(1) and 3(2);" "Figure 4" references to "Figures 4(1) and 4(2);" "Figure 6" references to "Figures 6(1) and 6(2);" "Figure 7" references to "Figures 7(1) and 7(2);" "Figure 8" references to "Figures 8(1) and 8(2);" "Figure 9A" references to "Figures 9A(1) and 9A(2);" "Figure 10" references to "Figures 10(1) and 10(2);" and "Figure 12" references to "Figures 12(1), 12(2), 12(3), 12(4), and 12(5)."

Rejection of Claims 1-4 Based on 35 U.S.C. § 103 and Richek

In the original patent application (serial no. 08/039,949), the Examiner rejected claims 1-4 based on the same grounds used to reject another claim (i.e., claim 6 allowed in a continuation patent application serial no. 08/484,947). In support of the rejection, the Examiner stated in paper no. 4 of the original patent application:

> "Richek et al. taught the invention substantially as claimed, including a data processing ("DP") system (as example in claim 6) comprised:
>
> a method of configuring a system in a computer (e.g., see the abstract);
>
> providing structural model hierarchy having structural relationships (e.g., see col. 4, lines 27-63, col. 7, lines 29-64, col. 21, lines 28-50, col. 22, line 58-col. 25, line 34, claims 9-11);
>
> providing a configuration instance (e.g., see col. 4, lines 27-46) and modifying the instance in response to a request by creating a model based on the request;
>
> storing the modification as a list (e.g., see col. 47, lines 7-10);
>
> examining said instance to determine if a constraint (conflict) exists (e.g., see col. 42, lines 42-44); and resolving/satisfying the conflict/constraint when exists (e.g., see col. 42, lines 45-50);"

Regarding claim 1, the Examiner states:

> "since [this claim is] an obvious modification of claim 6, [claim 1 is] rejected based on the rejections of claim 6."

Applicant respectfully disagrees. Applicant contends that for at least the following reasons Richek does not teach, suggest, or describe the method of claim 1:

1. *Richek does not teach, suggest, or describe the step of creating as in claim 1;*

2. *Richek does not teach, suggest, or describe an instance as in claim 1;*

3. *Richek does not teach, suggest, or describe structural relationships between elements of an element model; and*

4. *Therefore, Richek does not teach, suggest, or describe the method of claim 1, and claims 1-4 are patentable over Richek.*

The following provides a discussion of these reasons:

1. Richek does not teach, suggest, or describe the step of creating as in claim 1.

As stated in Richek (at col. 2, lines 25-29):

> "[t]he present invention is directed toward a method and apparatus used to allocate common computer resources to circuit boards installed in the computer system and to resolve conflicts which may arise in the allocation of resources."

Thus, Richek describes a method and apparatus that allocates resources. Richek describes a method wherein an integrator (a user of the method described in Richek) identifies the boards to be incuded in a computer system before invoking the method described in Richek. The integrator formats a diskette that contains a configuration file for each of the boards. Once the user has formatted the diskette with the configuration files for the boards for which resources are to be allocated, the method of Richek is invoked to configure the boards. According to Richek, the configuration process allocates resources for the boards identified by the configuration files that were placed on the diskette by the integrator.

According to Richek (at col. 36, lines 33-36):

> "[t]he board configuration file diskette contains all of the configuration information and files for boards installed or to be installed in the system."

In Richek, the determination of what boards are included in a computer system occurs before the configuration method is performed. As stated in Richek (col. 37, lines 28-37):

> "[w]hen the integrator has stored on the board configuration file diskette a configuration file or information for all boards that the integrator wishes to configure in the computer system, the integrator selects the configure option and control proceeds to step 500 (FIG. 4) where the configuration option actually commences. Based on the stored board configuration files, the system configures the options in step 502 by using the configuration method as described in the ALLOCATE, PROCESS and BACKTRACK subroutines 1100, 1200 and 1300."

Thus, the integrator determines the boards that are to be configured in Richek prior to the configuration process. The method in Richek merely allocates resources for the boards that the integrator includes in the system. That is, Richek does not teach, suggest, or describe the generation of a configuration for a system whereby components are created. Components are not created in Richek. In Richek, the boards that are to be part of the computer system are determined prior to the configuration process.

In contrast to Richek, the method of claim 1 does generate a configuration wherein components are created. Unlike Richek, the claimed method creates components during configuration. Thus, Richek does not teach, suggest, or describe the step of creating as claimed in claim 1.

2. Richek does not teach, suggest, or describe an instance as in claim 1.

The Examiner further states that:

> "Richek et al. did not specifically detail instance or constraint, exactly as claimed in the instant application. However, it would have been obvious to a person of ordinary skill in the art, at the time the claimed invention was made, Richek's configuration file statement including fields is the same as the claimed instance, and the claimed constraint is the same as Richek's conflicts."

Applicant agrees with the Examiner that Richek does not describe an instance as claimed. Given the Examiner's position that Richek does not detail an instance exactly as claimed, Applicant contends that it is incongruous to say that it would have been obvious to one of ordinary skill

that Richek's configuration file statement including fields is the same as an instance. Applicant contends that Richek's configuration file is not the same as an instance.

As stated in Richek (at col. 4, lines 47-50):

> "[t]he information in the configuration file consists of a series of parameters which serve two general purposes: common computer system resource allocation and circuit board initialization.

Thus, in Richek, a configuration file is merely a series of allocation and initialization parameters.

In Richek, a resource parameters identify the common system resources used by a circuit board as well as alternative resources that the board can use. The resource information is used to ensure that resources used by one board do not conflict with those used by other boards. As stated in Richek (at col. 4, lines 50-63):

> "Several parameters may specify common computer system resources used by a circuit board. These parameters may further specify various options for access to system resources that the board may use. For example, a file may contain the different number and type of interrupts that a board is capable of using. As described below, these parameters are used by the preferred embodiment of the present invention during the automatic computer system configuration process to ensure that the common computer system resources, such as memory address ranges, I/O address ranges, interrupt levels, and DMA channels used by a circuit board do not conflict with those other computer system devices."

Thus, in Richek, the resource parameter portion of a configuration file merely identifies the board's resource requirements. The resource parameters

are used during the configuration process to identify the resources used by the board.

The configuration file further contains initialization parameters. As stated in Richek (at col. 4, line 64-col. 5, line 13):

> "The second type of parameter concerns local circuit board specific operation alternatives; these parameters do not deal with common system resources. They determine how the board can be configured upon system initialization. For example, these parameters might include the baud rate, word length and parity selection for an asynchronous serial communications device. These parameters allow selection, at system configuration time, of the board operation alternatives which will be selected during initialization. The selected alternatives are then used to derive the information that the computer system initialization sequence uses to initialize the circuit board. For example, using these parameters, a memory board may be configured with portions of its memory partitioned among the conventional, extended and expanded memory areas available in products utilizing the operating system generally referred to as MS-DOS developed by Microsoft Corporation."

Thus, the initialization parameters of a configuration file merely define the initial settings for a board when the computer system in which the board is installed is booted (e.g., when the system is powered on).

Thus, according to Richek, a field in a configuration file contains resource parameters or initialization parameters associated with a specific board. A field in a configuration file in Richek is not the same as an instance of an element of an element model. That is, a field in a configuration file is not the same as an element in an element model as in claim 1.

3. Richek does not teach, suggest, or describe structural relationships between elements of an element model.

The Examiner states that Richek describes a model having structural relationships. The Examiner states:

> "Richek et al. taught the invention substantially as claimed, including a data processing ("DP") system (as example in claim 6) comprised:
>
> ...
>
> providing structural model hierarchy having structural relationships (e.g., see col. 4, lines 27-63, col. 7, lines 29-64, col. 21, lines 28-50, col. 22, line 58-col. 25, line 34, claims 9-11);"

Applicant respectfully disagrees. Applicant contends that Richek does not describe a model consisting of structural relationships between elements. None of the cited portions of Richek describe an element model that includes structural relationships between elements of the model.

The Examiner cites col. 4, lines 27-63 of Richek in support of the notion that Richek describes structural relationships between elements. Applicant respectfully disagrees. The cited portion of Richek describes a configuration file that consists of a series of parameters. These parameters specify a board's resource and initialization information. As stated in Richek (at col. 4, lines 47-50):

> "[t]he information in the configuration file consists of a series of parameters which serve two general purposes: common computer system resource allocation and circuit board initialization."

Thus, a configuration file consists of a series of parameters that are used to allocate resources and initialize the circuit board at startup. The cited portion of Richek further states that parameters can be contained in and accessible from a database or collection of configuration information or files. The description of a configuration file provided in Richek does not teach, suggest, or describe a model comprising structural relationships between elements. A series of resource and initialization parameters is not the same as structural relationships between elements in an element model.

The Examiner further cites Richek at col. 7, lines 29-64 in support of the position that Richek describes a model having structural relationships between model elements. This portion of Richek describes the contents of a vendor-supplied program that is used to initialize, change, update, and undo board information. According to Richek, a vendor-supplied program contains the following five executable modules: table program, initialization, change, update, and undo executable modules. The table program module provides a table of starting addresses for the remaining modules and data block pointers to the configuration means. Thus, the table program module consists of addresses and pointers used by executable modules in the vendor-supplied program. The initialization module is executed before the configuration means tries to configure the system. The change, update and undo modules provide executable code to change a function or choice, save configuration information, and undo a change, respectively.

Thus, the cited portion of Richek describes the format for a vendor-supplied program which is merely executable code or program. The

vendor-supplied program in Richek is not the same as an element model comprising structural relationships between model elements.

The Examiner also cites col. 21, lines 28-50 in support of the position that Richek describes structural relationships between elements of an element model. In the cited portion, Richek describes the initialization of a baud rate based on what port is selected as COM1 or COM2 and the storage on CMOS RAM of configuration information provided by a board manufacturer. There is no discussion or description of an element model comprising structural relationships between model elements. Resource and initialization parameters are not the same as structural relationships between elements in an element model.

Richek describes an array that is created from configuration files at col. 22, line 58-col. 25, line 34 and claims 9-11 also cited by the Examiner. According to Richek, the array contains an entry for each SUBFUNCTION, CHOICE, LINK, AND RESOURCE statements in a configuration file. An entry contains five fields. An entry includes a field that contains the statement in a configuration file for which the entry was created in the array. Two fields specify the status and type of a statement. A parent status field "is used to reflect the entry status and entry type of the statement which calls the present SUBFUNCTION, CHOICE, LINK, or RESOURCE statement" (col. 23, lines 56-59). A grandparent status field "reflects the entry status and type of the parent statement's parent" (col. 23, lines 60-61). Thus, the parent and grandparent status fields identify the type and status of a statement in a configuration file that calls another statement in a configuration file. The

grandparent and parent status fields relate statements in a board's configuration file. They do not provide structural relationships between elements of an element model.

Thus, according to Richek, an array stores configuration file statements. A statement is not an element. As discussed previously, a statement is used to specify resource and initialization parameters. An array is not an element model. There are no structural relationships in an array in Richek. Therefore, an array in Richek cannot be an element model consisting of structural relationships between model elements as in claim 1.

As can be seen from the discussion, the cited portions of Richek do not teach, suggest, or describe structural relationships between elements in an element model. There is no suggestion of a structural relationship between elements of an element model.

4. Therefore, Richek does not teach, suggest, or describe the method of claim 1, and claims 1-4 are patentable over Richek.

For at least the reasons set forth above, Applicant contends that Richek does not teach the method of claim 1. Thus, Applicant contends that independent claim 1 is not rendered obvious by Richek. Applicant therefore contends that claim 1 is allowable. Applicant further contends that claims 2-4 being dependent on an allowable independent claim are themselves allowable.

New Claims

Applicant has added new claims 16-29. Applicants contend that the specification, claims, and figures in the present application as originally filed provide support for these new claims. Applicant further contends that the new claims are patentable over Richek for at least the reasons set forth above with reference to claim 1.

Conclusion

For the foregoing reasons, Applicant contends that none of the references cited, either alone or in combination, teach, describe, or suggest the present invention. Applicant contends that pending claims 1-4 and 16-29 are allowable.

Respectfully submitted,

Hecker & Harriman

Date: 3/10/97

By: _____

J. D. Harriman II
Reg. No. 31,967

2029 Century Park East
Suite 1600
Los Angeles, CA 90067
(310) 286-0377
File No.: 85160.911CII

CERTIFICATE OF MAILING

This is to certify that this correspondence is being deposited with the United States Postal Service as Express Mail Label No. EM300602318US *in an envelope addressed to: Commissioner of Patents and Trademarks, Washington, D.C. 20231 on:*

March 10, 1997

Signature [illegible] *Date* 03/10/97